- Problem: Given a rectangle R containing a finite set S of n points belonging to R, partition R into a set $\mathcal{R}$ of rectangles such that each point in S is located on the boundary of a rectangle in the partition and the sum of lengths of boundaries of rectangles in $\mathcal{R}$ is minimized. $\leftarrow$ NP-hard

- A *guillotine cut* is a horizontal or a vertical line segment that cuts through a connected region and breaks it into at least two subregions. A rectangular partition is called a *guillotine rectangular partition* if it can be constructed by a sequence of guillotine cuts, each cutting through a connected subregion. We observe that there is an optimal guillotine rectangular partition such that it is a subset of a grid formed by drawing a horizontal and a vertical guillotine line segment through each point in S. A naive algorithm partitions R by using a guillotine cut so that R gets partitioned into two smaller rectangles and then it recursively partitions these smaller rectangles while using grid lines as guillotine cuts.

- An optimal (a minimum-lengthed) guillotine rectangular partition G of R that obeys points in S can be found in polynomial time.

  Proof: Considering the algorithm mentioned, there are $O(n)$ choices for any guillotine cut to choose. And there are altogether $O(n^4)$ possible subproblems, because each subproblem's boundary is composed of sections of $bd(R)$ plus at most four guillotine cut edges. Since, finding minima to fill each cell in the DP table takes polynomial time, this dynamic programming based algorithm essentially takes polynomial time and computes an optimal solution. Homework: provide all the details of a dynamic program.

- The minimum-length guillotine rectangular partition G is a 2-approximation to minimum-length rectangular partition OPT.

  Proof: Let us call each intermediate rectangle created by any sequence of guillotine cuts a *window*. The initial window is R itself.

  Here is a strategy to compute G from OPT: (i) if $int(W)$ does not contain any edge in OPT, then do nothing; (ii) if there is a *horizontal* line segment h in OPT that cuts through the whole window W, then apply a guillotine cut to W along h; (iii) if there is a *vertical* line segment v in OPT whose length is $\geq height(W)/2$, then apply a guillotine cut to W along v; (iv) if W contains at least one edge in OPT and neither (ii) nor (iii) is applicable, then apply to W a *horizontal* guillotine cut t that partitions W into two equal parts.

  In cases (i) and (ii), there is no additional cost (to OPT). In case (iii), the cost of guillotine cut is at most the length of v; charge this cost to v itself. We say a point z in a window W is said to be *chargeable* in W if each vertical half-line starting from z, but not including point z, going in either direction meets at least one horizontal line segment in $OPT \cap int(W)$. That is, we include a new horizontal edge t to $G \backslash OPT$ only if all of its points can be charged to edges in OPT that lie parallel to t. For case (iv), suppose a point z on t is not chargeable in W. Let R' be the rectangle defined by OPT that contains z. Since z is not chargeable, at least one edge of R' must abut an edge of W, leading to height of R' being at least $height(W)/2$, falling in case (iii). Otherwise, the width of R' is equal to the width of W, falling in case (ii). This contradiction facilitates charging half of t's cost to horizontal line segments in OPT that lie immediately above t and half of t's cost to horizontal line segments in OPT that lie immediately below t. Further, if a horizontal line segement s in OPT has been charged once by a new line segment t in G below it, then t becomes the boundary of the new window, and all points between t and s are non-chargeable in the new window containing s. In conclusion, each vertical line segment v in OPT is charged at most once, with cost at most $length(v)$, and each horizontal line segment h in OPT is charged at most twice, each time with cost at most $\frac{1}{2}length(h)$.

For any window W, a finite number of guillotine cuts suffices to partition W into rectangles so that for any rectangle T in those rectangles, $OPT \cap int(T) = \phi$. [Proof: in case (i), $OPT \cap int(W) = \phi$; cases (ii) and (iii) reduce the number by at least one in the interior of each of the subwindows they form; since t is chargeable in case (iv), the number of segments of OPT in either of the subwindows reduce from W.]

Implying, $OPT \subseteq G$.

In particular, G covers every given point in S.

Since we had shown how to compute an optimal guillotine rectangular partition in polynomial time and since G is a guillotine rectangular partition, proving G is a 2-approximation to optimal rectangular partition suffices. □

A PTAS for this problem was designed with further generalizations to guillotine cut definition.

References:

- Guillotine Cut in Approximation Algorithms by X. Cheng, D.-Z. Du, J.-M. Kim, H. Q. Ngo, Technical Report, 1989.